



## **Gestão e Qualidade de Software**

### **Sistemas de Informação**

Integrantes do grupo:

Kayky Cerquiaro Prado - 822155538

Enrico Aguiar Vrunski - 82210618

Thiago Ferreira Lima Gonçalves - 824156179

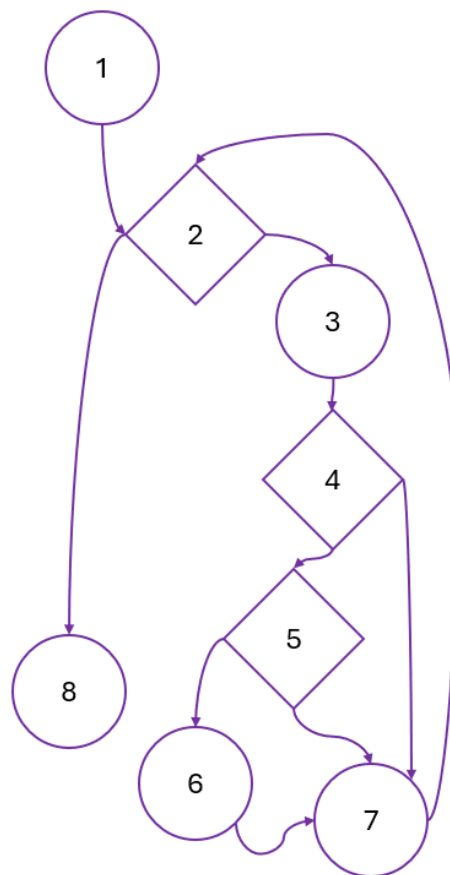
Matheus Tognon Siqueira - 824141731

Felipe Soares de Oliveira – 824156311

# Teste Caixa Branca

## Código com marcação de Nós:

```
1 ∨ public static int busca_binaria(int iVet[], int iK) {  
2     int iBaixo = 0, iAlto = iVet.length - 1;    // Nó 1  
3     int iMeio;  
4                                                     // Nó 1 (continuação)  
5 ∨     while (iBaixo <= iAlto) {                    // Nó 2 (condição do laço)  
6         iMeio = (iBaixo + iAlto) / 2;            // Nó 3  
7  
8 ∨         if (iK < iVet[iMeio]) {                  // Nó 4 (condição 1)  
9             iAlto = iMeio - 1;                    // Nó 5 (ação 1)  
10 ∨        } else if (iK > iVet[iMeio]) {            // Nó 6 (condição 2)  
11            iBaixo = iMeio + 1;                    // Nó 7 (ação 2)  
12 ∨        } else {                                  // Nó 8 (caso valor encontrado)  
13            return iMeio;                          // Nó 9 (retorno de sucesso)  
14        }  
15    }  
16  
17    return -1;                                     // Nó 10 (retorno de falha)  
18 }  
19
```



## Arestas (fluxos entre os nós):

Vamos contar os fluxos de decisão:

- De 1 → 2
- 2 → 3 (condição verdadeira)
- 2 → 10 (condição falsa)
- 3 → 4
- 4 → 5 (verdadeiro)
- 4 → 6 (falso)
- 5 → 2
- 6 → 7 (verdadeiro)
- 6 → 8 (falso)
- 7 → 2
- 8 → 9
- 9 → fim

Total de arestas (A): 12

## Complexidade Ciclomática:

$$C = A - N + 2C = 12 - 10 + 2 = 4$$

## Tabela de Casos de Teste:

Caso	Entrada(iVet)	Valor (ik)	Descrição	Saída esperada	Nós percorridos
CT01	[1, 2, 3, 4, 5]	3	Valor está no meio do vetor. Encontrado na 1ª iteração.	2	1 → 2 → 3 → 4 (F) → 6 (F) → 8 → 9
CT02	[1, 2, 3, 4, 5]	5	Valor está no final. Percorre à direita.	4	1 → 2 → 3 → 4 (F) → 6 (V) → 7 → 2 → 3 → 4 (F) → 6 (F) → 8 → 9

CT03	[1, 2, 3, 4, 5]	1	Valor está no início. Percorre à esquerda.	0	1 → 2 → 3 → 4 (V) → 5 → 2 → 3 → 4 (F) → 6 (F) → 8 → 9
CT04	[1, 2, 3, 4, 5]	6	Valor não está no vetor. Percorre até esgotar o while.	-1	1 → 2 → 3 → 4 (F) → 6 (V) → 7 → ... → 2 (F) → 10
CT05	[1, 2, 3, 4, 5]	0	Valor não está no vetor. Vai sempre à esquerda até sair.	-1	1 → 2 → 3 → 4 (V) → 5 → ... → 2 (F) → 10
CT06	[]	1	Vetor vazio. Não entra no while.	-1	1 → 2 (F) → 10
CT07	[1]	1	Vetor de um elemento. Valor encontrado direto.	0	1 → 2 → 3 → 4 (F) → 6 (F) → 8 → 9
CT08	[1]	2	Vetor de um elemento. Valor não encontrado.	-1	1 → 2 → 3 → 4 (F) → 6 (V) → 7 → 2 (F) → 10

## Teste de Caixa Preta

ID	Descrição do caso de teste	Entrada prevista	Saída prevista
CT01	Login e senha incorretos	Login: usuario123 Senha: senhaErrada	Exibir: <b>"Login e/ou Senha incorretos"</b>
CT02	Login correto, senha incorreta	Login: usuario123 Senha: senhaErrada	Exibir: <b>"Login e/ou Senha incorretos"</b>
CT03	Login incorreto, senha correta	Login: errado Senha: senha123	Exibir: <b>"Login e/ou Senha incorretos"</b>

CT04	Login e senha corretos, código errado	Login: usuario123 Senha: senha123 Código: 999999 (inválido)	Exibir: <b>"Login não autorizado!"</b>
CT05	Login e senha corretos, código certo	Login: usuario123 Senha: senha123 Código: 123456 (válido)	Exibir: <b>"Login realizado com sucesso"</b> e liberar acesso
CT06	Campos de login e senha em branco	Login:   Senha:	Exibir: <b>"Login e/ou Senha incorretos"</b>
CT07	Campo de código em branco após login válido	Login: usuario123 Senha: senha123 Código: ``	Exibir: <b>"Login não autorizado!"</b>
CT08	Código expirado	Login: usuario123 Senha: senha123 Código: 123456 (expirado)	Exibir: <b>"Login não autorizado!"</b>
CT09	Código inserido com atraso (fora do tempo limite)	Login e senha corretos Código inserido depois do tempo limite	Exibir: <b>"Login não autorizado!"</b>
CT10	Login e senha corretos, erro no envio de SMS	Sistema falha ao enviar o código por SMS	Exibir: <b>"Erro ao enviar código de validação. Tente novamente."</b>